Real-time Monitoring of Electromyography (EMG) using IoT and ThingSpeak

Dr. Mohd. Abdul Muqeet
Associate Professor, Muffakham Jah College of Engineering and Technology, Hyderabad, India.
(E-mail: ab.muqeet2013@gmail.com)

Abstract— Electromyogram (EMG) is a technique to record, evaluate and analyze the electrical activity produced by muscles. This technique is used to detect the issues that harm the nerves, muscle tissues and spot the location where they are joining. This paper discusses the implementation of a project which can be considered as a tool for the acquisition of muscle activity, presentation and real-time attainment of EMG signal using a specific EMG sensor. The live EMG reading is recorded using the Wi-Fi-enabled NodeMCU and Arduino and then sent to a remote server in our case ThingSpeak server with the help of IoT concepts which helps in the telemetry of the obtained biomedical signals using the cloud. The live recordings are also obtained on the PC using the Arduino serial plotter. This project can also help us in monitoring the progress of the patient treatment even if the physiotherapist could come and data can be directly sent to them. Thus the project aims to develop an EMG monitoring device based on IoT, for analyzing EMG signals.

Keywords— Internet of Things (IoT), EMG, NodeMCU, ThingSpeak.

I. INTRODUCTION

Electromyography (EMG) is a unique technique for specifying muscle activation. The area of EMG interpretation and pattern reorganization of biosignals have gained fast popularity during the past few years. This kind of research presents a smooth path to interface with the neuromuscular of handicapped people with the external world. The human body generates myoelectric signals using these signals the powered external device can be controlled. This process is referred to as myoelectric control (MEC) [1]. EMG signal is one of the important bio-signals which are generated by the human body, confirms the muscles activity or summation of various motor units action potentials [2]. EMG signals have the properties of non-stationary, nonlinear, complexity, and large variation. The Mio-Electric Signal (MES) is a complicated signal controlled by the central nervous system (CNS). It is affected by anatomical and physiological properties of muscles of the human body, the control scheme of the peripheral nervous system, and the characteristics of the instrumentation are used to detect and measure these signals.

EMG signal is normally a function of time and it is described in terms of the parameters amplitude, frequency, and phase. EMG signal also measures electrical impulses generated in muscles during its neuro-muscular activities. These electrical impulses are called action potentials which are responsible for muscle contraction. Surface electrodes placed on the skin above a muscle can measure these action potentials collectively also termed as EMG [3]. The EMG is a summation of all action potentials occurring in a muscle at a single time. Monitoring EMG has to lead to a greater understanding of muscle properties, giving insight into how muscles work together and coordinate tasks, and yield information about neuromuscular disorders [4]. This signal is complicated and non-stationary which is controlled by the nervous system because the nervous system is always responsible for muscle activity. The amplitude of the EMG signal is very small (50μv to 1mv) with frequencies varying from 10Hz to 3000Hz [5].

In [6] the authors extracted the time domain features from the EMG signal. The experiment was set up according to surface electromyography for noninvasive assessment of muscle (SENIAM). The recorded data were analyzed in the time domain to get the features. Based on the analysis, three features have been considered based on statistical features. The features were then been evaluate by getting the percentage error of each feature. The less percentage error determines the ideal feature. The results show that the extracted features of the EMG signals in the time domain can be implemented in signal classification [6].

The rest of the paper is organized as follows. The hardware component description to implement the project is explained in Section II. The software description to implement the project is presented in Section III. The project description is explained in Section IV. The experimental results are depicted in Section V and the paper is concluded in Section VI.

II. HARDWARE COMPONENT DESCRIPTION

Initially, the signals from the muscle are collected using the surface electrodes and EMG sensor. Then the signal goes to NodeMCU which belongs to an Arduino family which helps in monitoring and analysis. The live recording goes to the PC for viewing in the form of graphs and these graphs are obtained with the help of Arduino IDE [11].

The processed signal is the processed further stored in the cloud for the purpose of telemetry and graph can be obtained using ThingSpeak. Certain value limit is been set for the signal so if the signal potential crosses the limit the buzzer will give alert. The block diagram of the project is mentioned in Figure 1.

The major components for the implemented work are,
1. ESP8266 NodeMCU Board.
2. EMG Sensor.
3. Surface Electrodes

The block diagram representation of the implemented project work is shown in Figure 1.

**Figure 1. Block Diagram**

A. **ESP8266 NodeMCU**

The ESP8266 is the name of a microcontroller designed by Espresso Systems. The ESP8266 itself is a self-contained Wi-Fi networking solution offering as a bridge from the existing microcontroller to Wi-Fi and is also capable of running self-contained applications. This module comes with a built-in USB connector and a rich assortment of pin-outs. With a micro USB cable, you can connect NodeMCU devkit [12] to your laptop and flash it without any trouble using Arduino IDE [11]. It is also immediately breadboarded friendly. The pin configuration of the ESP8266 NodeMCU is depicted in Figure 2.

**Figure 2. Pin configuration of NodeMCU [12]**

B. **EMG Sensor**

As EMG activity measured in microvolts is linearly related to the amount of muscle contraction as well as the number of contracted muscles or in another word the stronger the muscle contraction and the higher the number of activated muscles, the higher the recorded voltage amplitude will be [7]. Muscle fibers generate electric activity whenever muscles are active. EMG signals are recorded by placing electrodes close to the muscle groups. Surface electrodes used in EMG recordings can either be “active” or “passive” electrodes. In the passive electrode type, the electrode consists of a simple silver/silver-chloride detection surface that senses current on the skin through the skin-electrode interface [8]. This type of electrode is normally used when the electromyography requires precise placement or if older EMG equipment in use. Active electrodes place a preamplifier either within the electrode or very close to the EMG data collection device. The advantages of surface electrodes are that there is minimal pain with the application, they are more reproducible, they are easy to apply, and they are very good for movement applications [9]. Surface electrodes are easy to apply and use, and they provide a good indication of muscle activity, with minimum discomfort to the subject. Once we have a clean EMG signal, we can begin to look at the data and try to figure out what it is telling us about the muscles [10]. The primary information to be gained is timing (on/off) information. In most movement analysis situations this timing information can be read directly from the raw EMG signal, no processing other than that which is used for cleaning up the raw signal (high and low-pass filters) is required [8]. The EMG sensor is depicted in Figure 3. This sensor will measure the filtered and rectified electrical activity of a muscle; outputting 0-Vs Volts depending on the amount of activity in the selected muscle, where Vs signifies the voltage of the power source. Power supply voltage: min. +3.5V [13]. This sensor will measure the filtered and rectified electrical activity of a muscle, depending on the amount of activity in the selected muscle. By detecting the electromyogram (EMG), measuring muscle activity has traditionally been used in medical research, however with shrinking but more powerful microcontrollers and integrated circuits advent EMG power Road and sensors can be used for various control systems. The sensor will measure the electrical activity of the muscle output 0-Vs volts, the output size to take Depending on the amount of muscle activity is selected [13].

**Figure 3. EMG Sensor**
C. Buzzer Module

A buzzer or beeper is an audio signaling device, which may be mechanical, electromechanical, or piezoelectric. The buzzer is an integrated structure of electronic transducers, DC power supply, widely used in computers, printers, copiers, alarms, electronic toys, automotive electronic equipment, telephones, timer, and other electronic products for sound devices. Typical uses of buzzers and beepers include alarm devices, timers, and confirmation of user input such as a mouse click or keystroke. The buzzer module used in our project is shown in Figure 4.

![Figure 4. Buzzer Module](image)

III. SOFTWARE DESCRIPTION

For implementing the code for the project, Arduino IDE is used which facilities, source code editing, program debugging, and complete simulation in one powerful environment. The Arduino IDE development platform is easy-to-use and helps you quickly create embedded programs. The IDE editor and debugger are integrated into a single application that provides a seamless embedded project development environment [16]. The most basic way to use the ESP8266 module is to use serial commands, as the chip is basically a Wi-Fi /Serial transceiver. The code for the project is compiled debugged and the .ino file is created. The method to download .ino File into Flash Memory of ESP8266 NodeMCU is also very easy and is done using the COM Port of computer PC. Figure 5 shows a view of the Arduino IDE.

![Figure 5. Viewing Program in Arduino IDE](image)

IV. PROJECT DESCRIPTION SETUP

Initially, the electrodes are placed on the muscles and the signals are collected using the surface electrodes and EMG sensor. Then the signal goes to NodeMCU which belongs to an Arduino family which helps in monitoring and analysis. The EMG readings are collected using the P.C for viewing in the form of graphs and these graphs are obtained with the help of Arduino IDE [11]. The processed signal is the processed further stored in the cloud for the purpose of telemetry and graph can be obtained using ThingSpeak. Certain value limit is been set for the signal so if the signal potential crosses the limit the buzzer will give alert. In order to obtain the measurement and monitoring of EMG online readings, the following description explains the project setup.

Download Arduino IDE from Arduino.cc with version 1.6.4 or greater. You can also try downloading the ready-to-go package from the ESP8266-Arduino project if the proxy is giving you problems. Install the ESP8266 Board Package Enter http://arduino.esp8266.com/stable/package_esp8266com_index.json into Additional Board Manager URLs field in the Arduino v1.6.4+ preferences.

ThingSpeak server is an open data platform and API for the Internet of Things that enables you to collect, store, analyze, visualize, and act on data from sensors [14]. ThingSpeak is a free web service that lets you collect and store sensor data in the cloud and develop Internet of Things applications. The ThingSpeak web service provides applications that let you analyze and visualize your data in MATLAB, and then act on the data. ThingSpeak is an open data platform for the Internet of Things [14]. Our project application can communicate with ThingSpeak using an API and we can either keep our data private or make it public. We have used the ThingSpeak to analyze and act on our data. This information can be in numerical values can be monitored live on ThingSpeak which can be viewed globally by the patient or his/her near and dears.

![Figure 6. The layout of ThingSpeak server Settings](image)
Next, use the Board manager to install the ESP8266 package. Click ‘Tools’ -> ‘Board’ -> ‘Board Manager’ to access this panel. Scroll down to ‘esp8266 by ESP8266 Community’ and click “Install” button to install the ESP8266 library package.

Once installation completed, close and re-open Arduino IDE for ESP8266 library to take effect. When you’ve restarted Arduino IDE, select ‘NodeMCU 1.0’ from the ‘Tools’ -> ‘Board’ dropdown menu. Go to your Windows ‘Device Manager’ to find out which Com Port ‘USB-Serial CH340’ is assigned to. Select the matching COM/serial port for your CH340 USB-Serial interface. As shown in Figure 8 the board and port are selected using the Arduino IDE.

V. SYSTEM RESULTS

The experimental setup is depicted in Figure 9. This system has been tested by placing the surface electrodes on a human body for reference, detection and ground junction.

Specifically, Figure 10 depicts the placement of electrodes on the muscles group and Figure 11 describe the placement of the electrode on the left leg to create reference junction. These electrodes are connected to the EMG sensor which when supplied with power start collected data.
When this EMG sensor is interfaced with the NodeMCU then after compiling and building the code which also configures the ThingSpeak server we can also monitor the live and real-time EMG readings. This implementation thus utilizes the concept of IoT. The EMG monitoring signal is displayed on the ThingSpeak server as depicted in Figure 12.

![Figure 12. ThingSpeak Graph](image)

We can plot the real-time EMG signals using the Arduino IDE using its Serial Plotter function as shown in Figure 13. Of course, the EMG sensor and NodeMCU are interface and connected to a PC with USB interface.

![Figure 13. Arduino Serial Plotter](image)

Thus implementing the given project we can obtain real-time readings of the EMG using the EMG sensor and Wi-Fi enabled NodeMCU. Utilizing the IoT concept the data can be transmitted to ThingSpeak server. This data can be visualized privately or publicly as per the requirement. We offered a low-cost monitoring of the EMG monitoring system.

VI. CONCLUSION

This work explains the implementation of IoT based real-time monitoring of EMG sensor data using the EMG sensor and NodeMCU. The project implements a very cost-effective biomedical system using the Arduino IDE and required hardware. Additionally, the IoT implementation also sends the local EMG data readings to the ThingSpeak server which could be monitored from anywhere in the world. This project opens the areas of real-time acquisition and real-time processing of EMG data. This project can also help us in monitoring the progress of the patient treatment even if the physiotherapist could come and data can be directly sent to them. Thus the aim to develop an EMG monitoring system based on IoT for analyzing EMG signals can be well established here.

REFERENCES